

# PROV Data Model - RDF Interoperability – Data Properties, Object Properties, UML Attributes, UML Associations

Luc Moreau<sup>1</sup>

University of Southampton, UK  
l.moreau@ecs.soton.ac.uk

**Abstract.** PROV-DM and PROV-O specifications make interoperable round trip conversions between RDF and PROV-N difficult.

## 1 Subject

Round-trip Interoperability between PROV-DM and RDF. Definitions of properties (prov:location, prov:role, prov:type) as attributes in PROV-DM and object properties in PROV-O.

## 2 Application

Problem identified in ProvToolbox round trip testing.

## 3 Background

PROV-DM [2] contains illustrative UML diagrams showing PROV classes and PROV associations. The UML diagrams are not normative, but the normative text describing the data model and the PROV-N [1] notation are compatible. UML associations correspond to PROV-N terms such as `wasGeneratedBy(id;e,a,t,attrs)` or `used(id;a,e,t,attrs)`. On the other hand, `prov:location`<sup>1</sup>, `prov:role`<sup>2</sup>, and `prov:type`<sup>3</sup>, are not defined as UML associations, but are class attributes.

In PROV-O [3], the corresponding properties `prov:atLocation`<sup>4</sup>, `prov:hadRole`<sup>5</sup>, and `rdf:type`<sup>6</sup> are defined as Object properties.

My interpretation for these distinct designs is the following. In PROV-DM, the intent was to allow programmers to provide a string for a role, location, or type, without having to create IRIs. For instance, a location could be expressed

<sup>1</sup> <https://www.w3.org/TR/prov-dm/#term-attribute-location>

<sup>2</sup> <https://www.w3.org/TR/prov-dm/#term-attribute-role>

<sup>3</sup> <https://www.w3.org/TR/prov-dm/#term-attribute-type>

<sup>4</sup> <https://www.w3.org/TR/prov-o/#atLocation>

<sup>5</sup> <https://www.w3.org/TR/prov-o/#hadRole>

<sup>6</sup> [https://www.w3.org/TR/rdf-schema/#ch\\_type](https://www.w3.org/TR/rdf-schema/#ch_type)

directly as the coordinate string. In PROV-O, the object of `prov:atLocation`<sup>7</sup> is an instance of `prov:Location`<sup>8</sup>. The motivation for this is that it could also be an entity with its own provenance: most countries have borders changing over time, countries are created, countries also cease to exist. Both design rationales seem reasonable to me, but ... they cause an inter-operability issue.

## 4 Problem To Address

How to convert:

- to an RDF representation, and vice versa,
- to PROV-DM,

and ensure proper round-trip conversion! Likewise for `prov:role` and `prov:type`. The problem does not seem to exist for `prov:value`.

## 5 Solution

It would be good to analyse how these properties are used in the wild, in the different representations, to see what programmers actually generate.

A number of options are possible:

- Provide conversion rules from PROV-DM values to some instances (denoted by a URI).
- Restrict the range of values in PROV-DM definitions to be Objects, but this implies that these properties become associations between classes in UML speak.
- Allow both data and object properties in all representations.

## 6 Solution Rationale

A solution is required to ensure inter-operable round trips, and agreed test suites.

## References

1. Luc Moreau and Paolo Missier. (2013). PROV-N: The Provenance Notation. W3C Recommendation.. <http://www.w3.org/TR/2013/REC-prov-n-20130430/>
2. Luc Moreau and Paolo Missier. (2013). The PROV Data Model. W3C Recommendation.. <http://www.w3.org/TR/2013/REC-prov-dm-20130430/>
3. Timothy Lebo, Satya Sahoo and Deborah McGuinness. (2013). PROV-O: The PROV Ontology. W3C Recommendation.. <http://www.w3.org/TR/2013/REC-prov-o-20130430/>

<sup>7</sup> <https://www.w3.org/TR/prov-o/#atLocation>

<sup>8</sup> <https://www.w3.org/TR/prov-o/#Location>